Analysis of Gene Expression Data **Spring Semester, 2005**

Lecture 3: March 12, 2005

Lecturer: Rani Elkon *Scribe: Amos Mosseri and Eitan Hirsh*[1]

3.1 Low level analysis of microarrays

3.1.1 Introduction

This course deals with *High level analysis* of data gathered by microarrays. Different types of High level analysis include:

- (Bi-)Clustering
- Reconstruction of transcriptional networks
- Induction of classification rules

All of these high level analysis methods are based on the same *raw data* - A numerical description of the expression level for a number of genes, along a number of experiments. *Low level analysis of microarrays* is the set of methods used to obtain this so called *raw data* from the physical data gathered from the microarray (see Figure 3.1), i.e. luminance measurements for each probe on the array.

The numerical values for expression levels should be extracted from the luminance levels, normalized, and systematic errors should be removed. All of these tasks are handled by Low level analysis of microarrays

3.1.2 Microarray technologies

There are two types of microarray technologies. single channel microarrays and dual channel microarrays

Single channel

Microarrays presented with a single type of target (eg. a treatment tissue)

[1]Based in part on a scribe by Erez Yaffe and Dan Cohen, March 2004

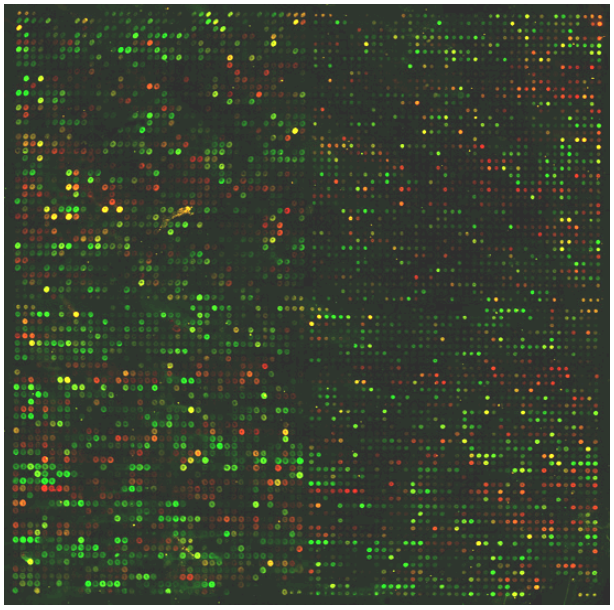

Figure 3.1: Scanned microarray result.

Dual channel

Microarrays presented with two different targets, usually used when comparing between the two (eg. test and control cells in the cDNA technology where the probes' quantity could be different from one chip to the other). (see Figure 3.2)

3.1.3 Types of microarrays

Currently, three types of microarrays are in widespread use:

- Spotted cDNA microarrays
- Spotted oligonucleotide arrays produced by *Agilent*
- GeneChip arrays produced by *Affymetrix*

Each of these microarrays will be now presented.

Spotted cDNA microarrays

In a spotted cDNA microarray, each probe is a mRNA sequence or an EST[2] created by the method of PCR. The probes' length is 300-1000 bases. The probes created by PCR are

[2]ESTs are mRNA sequences that form a fraction of a gene's sequence

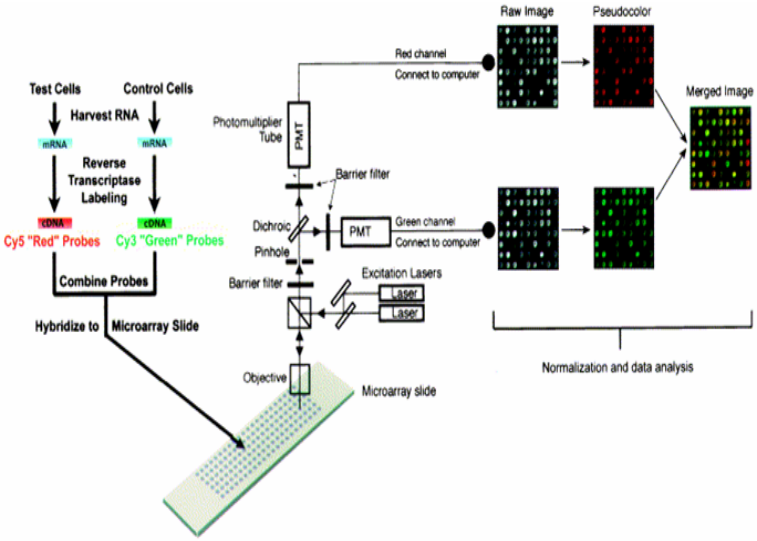

Figure 3.2: Dual channel technology. Notice the use of two different colors on the same chip, during the preparation process and analysis process.

double stranded, so heat is used to separate them before they are placed on the array. The probes are placed on the chip using a *spotter*, which is a mechanic head that touches test tubes containing the probes and then touches the microarray, placing the probes on it, (see Figure 3.3). The chips are created in batches of 200, (see Figure 3.4)

The Spotted cDNA microarray suffers from the fact that not all of the probes are single stranded (there is no way to know how many of the probes were separated) thus hindering hybridization, and very long, thus causing *cross-hybridization* in which a target binds with a probe, even though it does not match it completely. On the other hand, this is a relatively cheap method to create microarrays and used primarily by research facilities[3] to create their own chips. About 50 percent of the microarrays used nowadays are *Spotted cDNA microarrays*. An additional advantage of this method is in the special case of finding difference between two samples, whether expressed in known genes or unknown ones. After all, we don't always want to search for the coin under the flashlight.

Spotted oligonucleotide arrays

Spotted oligonucleotide arrays are created by Agilent and use synthetic oligonucleotides as probes. Each probe is 60-70 bases long and placed on the chip using an inkjet printer

[3]Starting Stanford at 95'

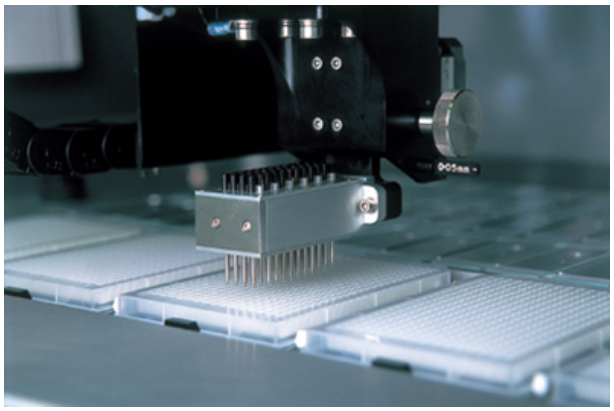

Figure 3.3: Mechanic head touches test tubes

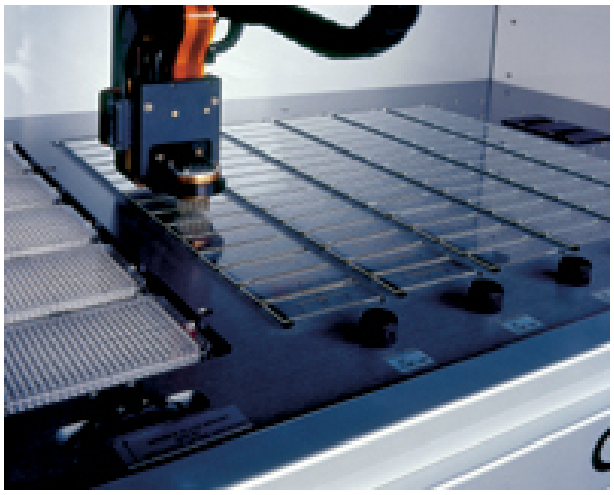

Figure 3.4: Create batch chips

(Agilent uses HP for the printing technology). Using synthetic oligonucleotides means that the probes are single stranded, with known sequence, thus allowing better hybridization and less cross-hybridization. On the other hand, this method is much more expensive.

The probes' sequences are chosen according to the purpose of the chip, i.e. the genes it is supposed to detect. There are a number of available types of chips:

- Human
  - **Whole human genome microarray: 44K probes (41K known and predicted genes**
  - **19k well characterized genes (1A)**
  - **19k ESTs and predicted genes (1B)**
- Mice - 41K probes representing over 20k genes
- Other organisms - Rat, Arabidopsis, rice, yeast

The method is quite new and thus not wide-spread as the other two. Around 5 percent of the microarrays used nowadays are Spotted oligonucleotide arrays.

Affymetrix GeneChip arrays

Affymetrix microarrays are currently the most commonly used commercial microarrays. In these microarrays, for each gene that a microarray is intended to detect, a number of probes(11-20) called *positive match probes (PM)* are set. These probes are about 25 bp long, matching different positions along the gene. Furthermore, for each such probe, another probe, a *mismatch probe (MM)* is placed on the microarray (see Figure 3.2). This mismatch probe is identical to the correct probe with exception of the base located in the middle. The mismatch probe is used to detect cross-hybridization, in which case the probe and its mismatch probe will both bind to the target. When hybridization occurs only for the correct probe, and not its mismatch probe, we know that this is true hybridization. (see Figure 3.5))

Let's assume for example that the sequence of the gene to detect is

ATGC**TGATC**GATGCAGAATCGATC

one possible (yet short) probe will be TGATC. The chips will contain this probe and also TGTTC. The possible hybridization results will be analyzed as follows:

- Both probes are detected - cross-hybridization or non specific binding has occurred. this probe won't provide any useful information.

- Only the correct probe is detected - a specific binding occurred. Of course, in a real experiment one would require all of the correct probes (or at least most of them) to be detected in order to decide that the gene is present.
- Neither probe was detected - the target gene probably isn't present.

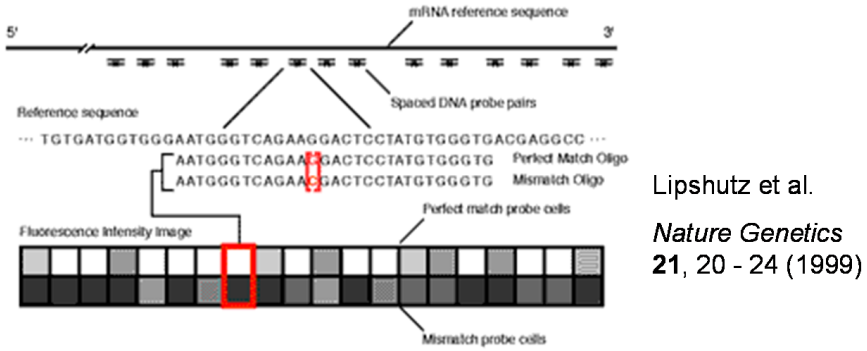

Figure 3.5: Affymetrix GeneChip arrays. An example of the PM-MM probe pair method.

Note that during the reverse transcriptase labelling (used to create the cDNA, (see Figure 3.2)) we can use only bases close to 3', up to a distance of about 700 bases. This enforces us to take all 11-20 probe pairs from this region.

As in Agilent chips, there are tailor made chips for a number of organisms:

- Human
  - **Human Genome U133 plus 2 - 47,000 probe sets for known genes and EST transcripts**
  - **Human Genome Focus Array: 8,500 well annotated genes**
  - **Human Cancer G110 Array: 1,700 genes implicated in cancer**
- Near whole genome chips - Rat, Drosophilae, C. elegans, Arabidopsis, Yeast, Zebra Fish, E. Coli
- Future plans
  - Tiling chips: coverage of the whole genome. Some researchers suspect that there is a biological meaning to the "trash" DNA sequences. For that reason Affymetrix decided to create microarray chips containing the whole human genome.
    - Currently available for Chr 21 and 22

    - 10 additional chromosome under construction
    - Used to discover: Transcribed MicroRNAs (Non-coding genes), TF binding sites and Sites of chromatin modifications
  - Exon chips: Identify splice variants (alternative splicing). In different tissue types (e.g. brain and eye tissues) there are different splicings for the same DNA sequences. These chips might help understanding the different splicings and thus protein production in different body cells.

Low level analysis - steps

The low level analysis is divided into three major steps, as follows:

- Image Analysis
- Signal summery (Affymetrix)
- Normalization

3.1.4 Image analysis

The first step in low level analysis of a microarray is *image analysis*, a process in which the raw visual data of observed illumination intensities is transformed into an estimate for gene expression levels (for each probe). This step is mostly composed of image processing tasks, transformation of the image into numbers.

Grid alignment

Before extracting the intensity for each probe, one has to locate it. In order to locate the probes one has to superimpose a grid on the scanned intensity levels picture, thus finding the border of each probe. Unfortunately there are many error factors (e.g. movement of the scanner during the scan) which make it hard to align a grid with the entire picture. This can be solved by segmenting the picture and aligning each segment to its own grid (see Figure 3.6). Affymetrix microarrays are always created with E-coli probes along their border. By adding E-coli to the tested sample, one can make sure that these probes will be lighted (i.e. detected) and will help determine the border of the chip, and its grid alignment (see Figure 3.7).

Target detection

Target detection is the process of deciding which pixels in the scanned picture will be used to calculate the intensity of a probe. This task is especially important in Spotted cDNA microarrays in which the spotter creates an uneven spread of each probe's copies, thus

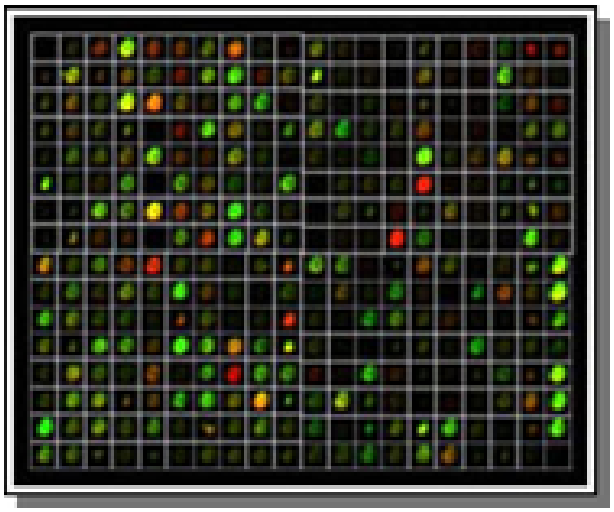

Figure 3.6: general grid alignment. [12]

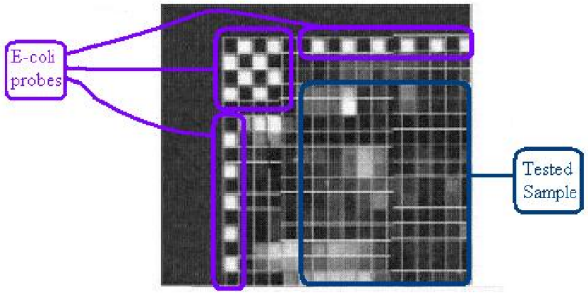

Figure 3.7: Affymetrix chip grid alignment - An example of illuminating the corner and borders of the array.

creating an uneven intensity measurement for each probe type (see Figure 3.8) and (see Figure 3.9).

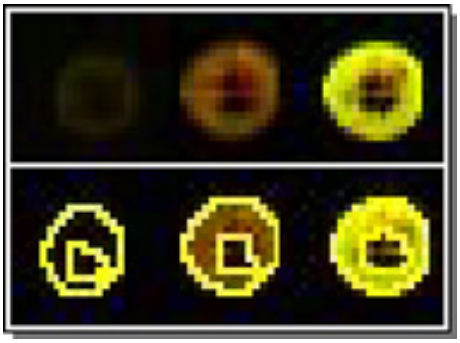

Figure 3.8: Target detection. Notice the highlighted pixels the target detection method locked on.

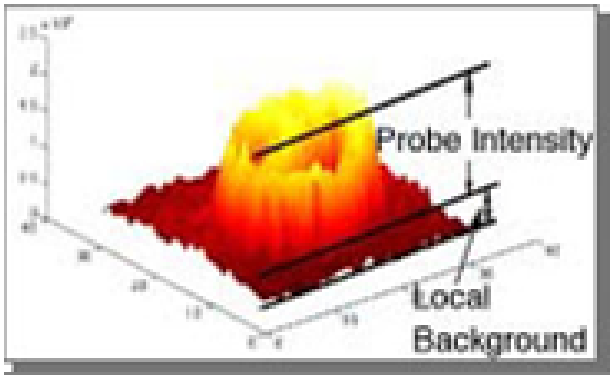

Figure 3.9: Intensity picture for cDNA micro as a function of the grid cell's pixel location. Notice the crater like distribution of probes. [12]

Target intensity extraction

Given all of the relevant pixels for a probe, one needs to compute a numerical value representing the expression level for that probe. This could be the mean intensity value, the median, etc.

Affymetrix use a 64 pixel per cell resolution and take the 75th percentage as the cell's value, dismissing border pixels. (see Figure 3.10)

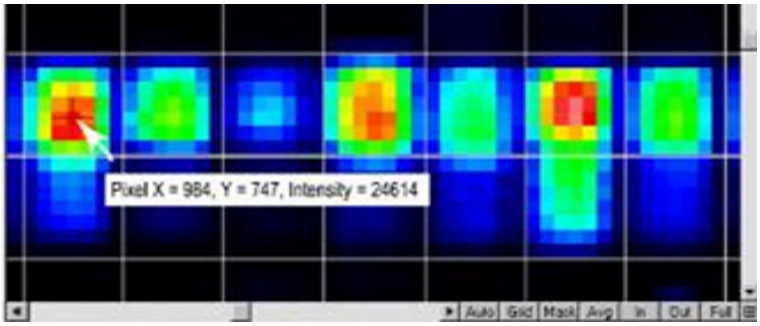

Figure 3.10: Affymetrix target intensity extraction. Notice the different expression level in the PM and MM cells.

Local background correction

The intensities measured may be severely biased due to dust, glare and non specific binding. Local background correction is used to crudely correct these biases.

3.1.5 Summation of probe set signals for Affymetrix chips

As was explained before, an Affymetrix chip has a number of probes for each gene it intends to detect. In this step, needed only for Affymetrix chips, one computes a numeric expression estimate for the gene, based on the expressions values for each correct probe (PM) and mismatch probe (MM). There are a number of methods to do this calculation ([8],[5],[6]).

In general we will mark the expression level of probe j for gene i by index $_{ij}$. The expression level for positive probe j will be marked as PM_{ij}, the expression level for mismatch probe j will be marked as MM_{ij}, the *true* expression level for gene i will be marked θ_i, the calculated expression level for gene i will be marked E_i.

Average Difference (MAS 4)

This method is based on the idea that the gene expression level is estimated by the difference between the PM and the MM value, with the exception of completely random error :

$$\theta_i + \epsilon_{ij} = PM_{ij} - MM_{ij}$$

Thus, to cancel the noise we should take the mean value for all of the probes :

$$E_i = \frac{\sum(PM_{ij} - MM_{ij})}{T}$$

(T is the number of MM-PM probe couples).

A possible improvement is to ignore outliers - probes with intensities very different from the rest and treat them as measurement errors.

The problem with the MAS4 model is that it assumes ϵ_{ij} has equal distribution so it could be cancelled by a simple mean. It appears that the distribution of errors depends on the general intensity of the probe, and its mean value increases with the targets' expression levels. (see Figure 3.11)

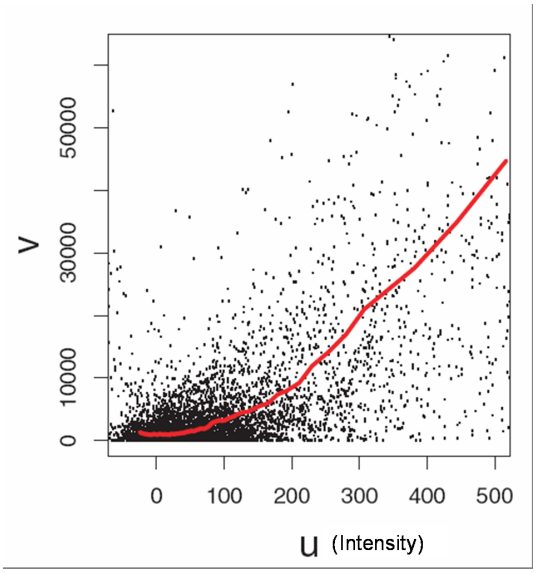

Figure 3.11: Error level growing with the intensity.

MAS 5

One way to reduce this dependency is to use a different error model, a multiplication error factor

$$PM_{ij} - MM_{ij} = \epsilon_{ij} \cdot \theta_i$$

which can be transformed using log to give us

$$log(PM_{ij} - MM_{ij}) = log(\epsilon_{ij} \cdot \theta_i)$$

and

$$log(E_i) = \frac{\sum(log(PM_{ij} - MM_{ij}))}{T}$$

Again, in order to handle obvious measurement errors, one could give a smaller weight to values far from the mean (in comparison to the values' variance), i.e. use

$$log(E_i) = \sum(w_j \cdot log(PM_{ij} - MM_{ij}))$$

When w_j is bigger when PM_{ij}, MM_{ij} are closer to their mean.

dCHIP

The dCHIP method, devised by Li and Wong ([14]), is based on a model in which in addition to random errors each probe has a different affinity to hybridization, i.e. some of the probes for the same gene have stronger affinities and will be more expressed (see Figure 3.12).

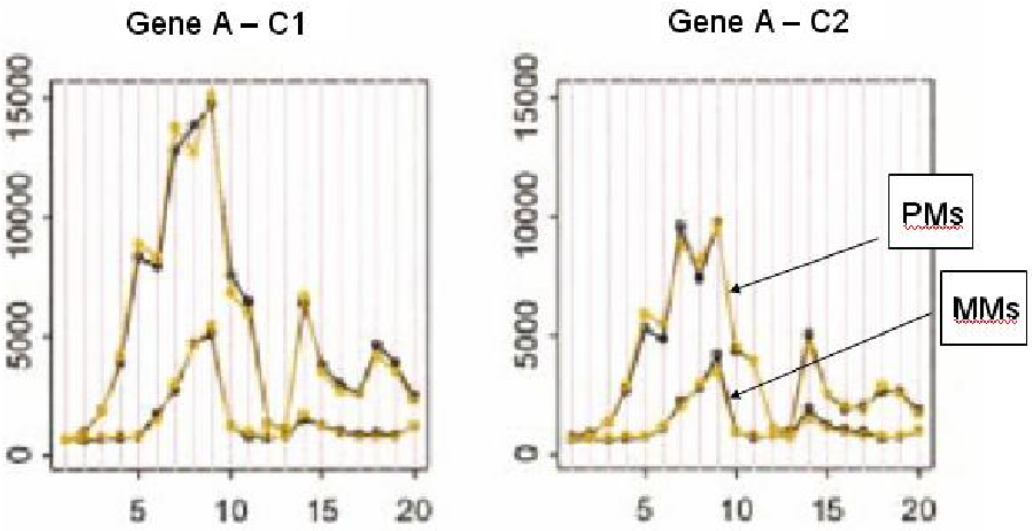

Figure 3.12: Affymetrix probes affinity effect. The X-axis represents the probe pair serial number. The Y-axis represents the genes expression levels. Two different conditions are displayed here, C1 and C2. [6]

$$\alpha_j \cdot \theta_i + \epsilon_{ij} = PM_{ij} - MM_{ij}$$

when α_j is the affinity of probe j to hybridization. Multiple arrays (10-20) are required in order to fit a model and obtain good estimates for α_j and θ_i. This can be done once for every kind of chip[4].

Robust Multi-array Average(RMA)

This method is based on the idea that the MM values are strongly dependant on PM values, (see Figure 3.13). Thus cannot be used to improve results based solely on PM values. The

[4]Chips with the same probes

model used here is based on the multiple error factor as MAS5 and different affinity levels as in dChip, in addition to ignoring the MM values:

$$log(\alpha_j \cdot \theta_i) + \epsilon_{ij} = log(PM_{ij})$$

θ_i is estimated by using a robust linear fitting procedure.

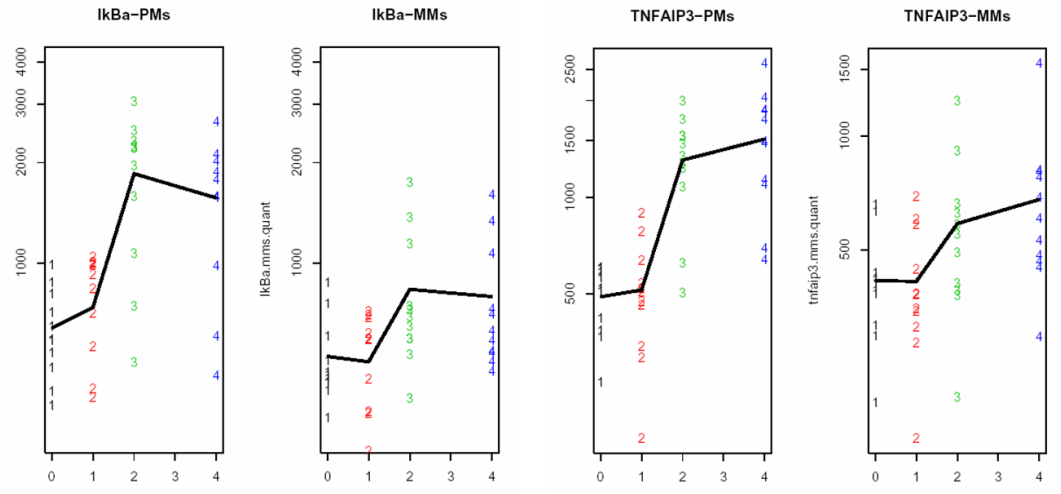

Figure 3.13: The correlation between the PM and the MM values of gene IkBa (The Left figures)and TNFAIP3 (The right figures). The X-axis is the probe pair number. The Y-axis is the median expression level.

PLIER - Probe Logarithmic Intensity Error Estimation

The method Affymetrix are using in their new chips.

- Take affinity into consideration - Affinity estimates have been generated using a large experimental dataset across multiple tissues
- Error model smoothly passes from additive (at low intensities) to multiplicative
- Input for model fitting: PM - MM, PM - B[5], PM, PM + MM

[5]Background intensity

Comparing the methods

To compare the effect of different methods, a controlled test was performed. 11 known RNAs were added to a test sample in known concentrations (which were much higher than the concentrations of native sequences in the sample). The expression levels of these RNA samples were calculated using each of the methods and the results were compared to the correct values. Based on these tests, it appears that RMA is the best among the presented methods (see Figure 3.14).

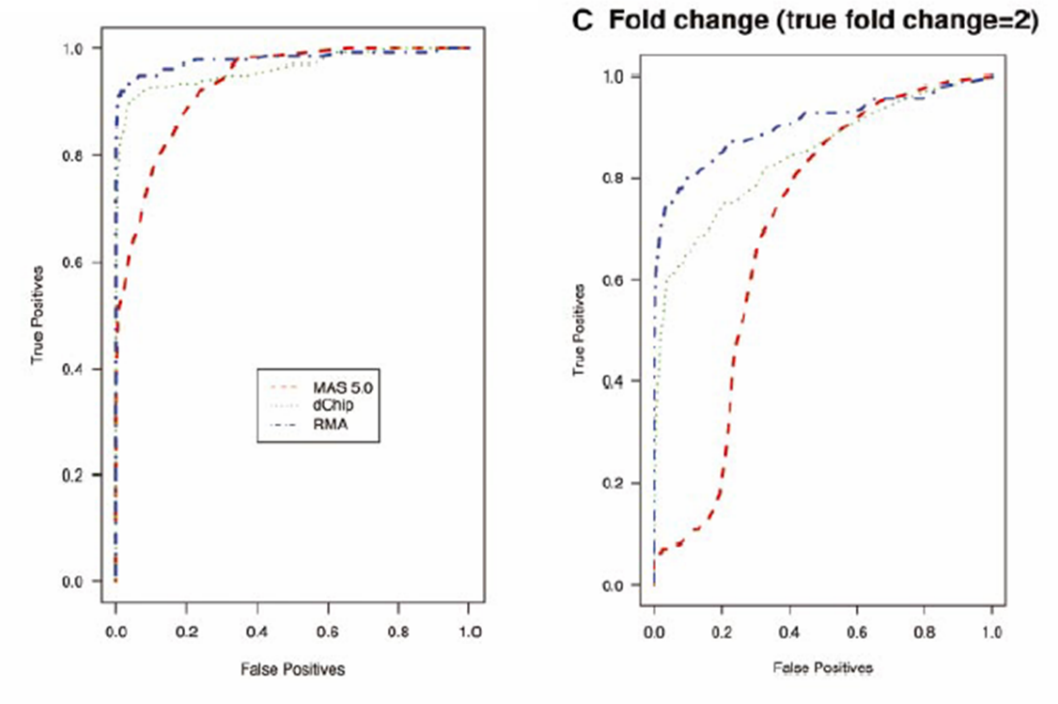

Figure 3.14: Comparison results: RMA, dChip and MAS5. The left figure depicts the experiment results when the spiked-in RNA's were selected randomly. The right figure depicts the results of spiked-in RNA's with concentrations 2 times higher than those of the test sample. Notice that these are ROC diagrams. The closer the curve is closer to the upper left corner, the better is the performance. [6]

3.1.6 Normalization

The normalization step is intended to deal with the fact that the results from identical experiments on two identical microarrays will never be exactly the same. In addition to

unavoidable random errors (see Figure 3.15A) there are also systematic differences (see Figure 3.15B) caused by:

- Different efficiencies of dyes. For example, green colored markers are stronger then red ones (measured as stronger illumination) thus creating a bias between experiments done with green and red markers.
- Experimental differences (whether by mistake or because of differing experimental protocols) will lead to different amounts of mRNA in the tested sample, causing different expression levels. This problem is especially important when comparing data gathered in different laboratories.
- Different scanning parameters
- Differences between chips created in different production batches.

These differences can be corrected by the use of *normalization* methods which are the process of removing systematic errors (biases) from the data. Without correcting these differences, it is be impossible to compare the results of two experiments. In the following graphs, the gene

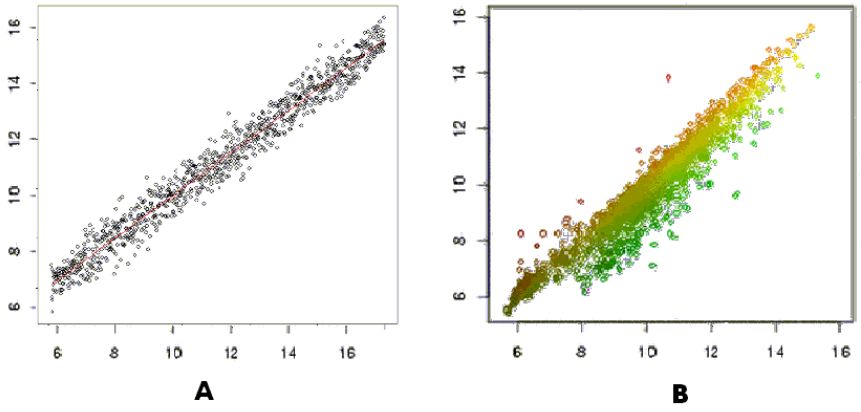

Figure 3.15: A comparison of two identical samples on different chips. The X-axis and the Y-axis are the intensity levels at each experiment. (A) shows expected results with noise, and (B) shows results with systematic bias. Ideally, all the data points will be on the main diagonal.

expression levels will be presented as a histogram of $log(intensity)$ values. The results from two chips (or two tests of the same sample with differing markers) will be colored in red and

green (e.g. see Figure 3.16A). Notice that even though a comparison of identical samples is used in Figure 3.16A, normalization is important when comparing different samples in order to detect differential genes. In such cases it is harder to normalize the results because one cannot know whether the different expression levels are caused due to actual differences or a normalization problem. A normalization scheme should answer two questions:

- Which genes (probes) are used for the normalization process
- How is the normalization performed, i.e. what is the mathematical algorithm used to normalize the values.

Finding the normalization genes

There are a number of methods for choosing the normalization genes, i.e. those genes on which the normalization scheme will be based.

1. All gene normalization

Using all of the genes on the chips for normalization is based on the assumption that most of the genes have the same expression levels in the two (different) samples which are compared. The proportion of the differential genes is low (less than 20 percent). Thus we can assume that by using all of the genes, we will have a large number of equally expressed genes for the normalization and achieve good results. This method cannot be used when the previous assumption is wrong. For example, when the samples are highly heterogeneous (e.g. samples from completely different tissues). Or when using dedicated chips (e.g. Human cancer arrays)

2. Housekeeping genes only

The idea is to use a small set of genes that, based on prior knowledge, are known to have equal expression levels in the compared samples. Two currently used normalization schemes are based on housekeeping genes:

- Affymetrix chips have a set of 100 housekeeping genes used for normalization
- NHGRI's cDNA microarrays have a set of 70 housekeeping genes

One problem with using housekeeping genes is that they are usually expressed at high levels, so they are not informative for the normalization of the low intensities range. Another problem is that the validity of the assumption about these genes equal expression level is questionable.

3. Spiked in controls

In the *spiked in controls* method, a number of control mRNAs are added to each sample. These mRNAs are taken from another organism (as to make sure that they do not exist in the sample itself). The microarrays are designed to have probes that detect these mRNAs. The controls are added in a range of concentrations thus providing normalization data for different expression levels. This method's main limitation is that due to the fact that the controls are added only to the final sample, they cannot compensate for differences caused during its preparation. Only differences in the scanning and image analysis steps can be compensated. Imagine two samples that were produced with different amounts of genes due to some experimental error. Later, the controls are added in equal amounts, so they can provide no clue on the initial difference. One should remember that sample preparation is probably the most common cause for biases, rendering this method much less effective. Furthermore, spikes normalization is based on small (70-100) number of probes so it isn't as robust as the other methods.

4. Invariant set

Contrary to the other methods, in the *Invariant set* method, one decides on the normalization genes only after the results are analyzed. The idea is to detect genes with similar expression levels in all of the chips, assume they should have identical expression level and base the normalization scheme on them. One way to detect these genes is by ranking the expression levels for all of the genes and choose genes with the same rank (global biases should have less effect on the comparative rank of each gene).

Normalization methods

Once the normalization genes were chosen, there are a number of methods for the normalization itself. One should remember that all of these methods are always computed based on the expression levels of the normalization genes, and later the transformation is applied to the entire data set.

1. Global normalization (Scaling)

This normalization scheme is intended to equalize the mean value of expression levels. All of the values are multiplied by a constant which is the ratio between the mean expression level of the normalization genes in the two samples. The normalization factor k is

$$k = \frac{\sum(E_i^1)}{\sum(E_i^2)}$$

when the summation is other normalization genes. (E_i^j is the expression level for gene i in sample j). Normalization of E_i^2 values is done by multiplication by k. (see Figure 3.16 and Figure 3.17). Note this this normalization will work only when we are considering a constant difference between the samples.

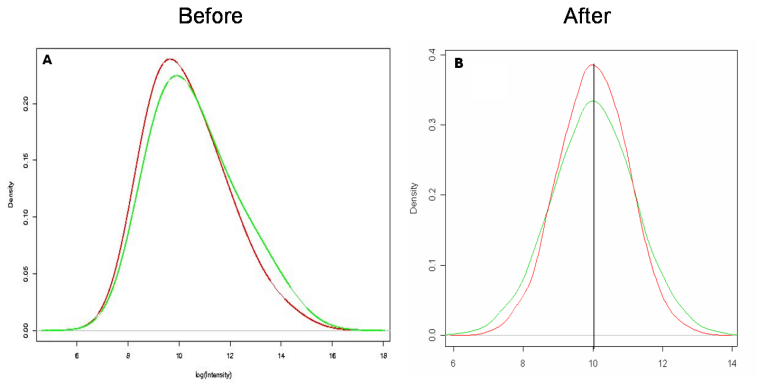

Figure 3.16: Histogram before (A) and after (B) global normalization. Here we see a distribution diagram of two identical samples when tested on two chips. The X-axis is the intensity level and the Y-axis is the density. After normalization, the mean of the two distributions is identical, although the distribution is not identical.

2. Lowess normalization - local linear fit

Lowess normalization([10],[7]) is based on the (true) assumption that the biases are intensity dependent, thus there is no one normalization factor that can remove the biases for higher and lower expression genes. One should normalize different expression level genes with different factors. Before tackling the Lowess normalization, it is important to be familiar with the $Mvs.A$ plots which help detect intensity dependent biases. The X axis is the average intensity of a gene in both samples(chips):

$$A = \frac{log(E_i^1 \cdot E_i^2)}{2}$$

The Y axis is the log ratio of these intensities:

$$M = log(\frac{E_i^1}{E_i^2})$$

For example, Figure 3.18 shows a situation in which there is no intensity dependent bias (the ratio between expression values (Y axis) does not change according to the expression

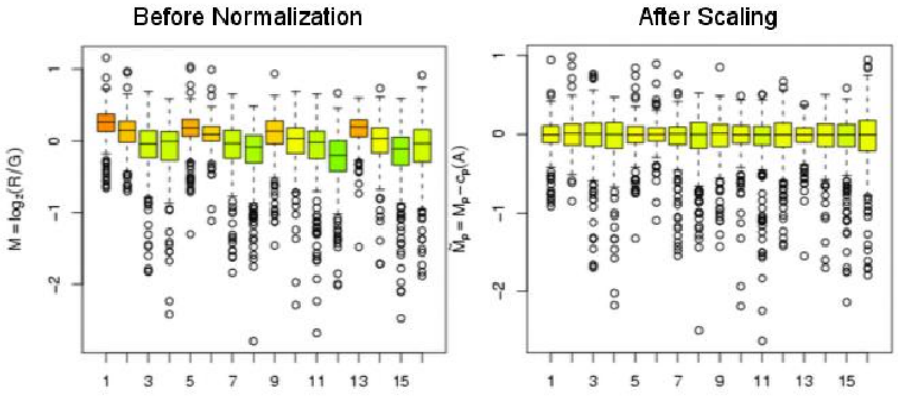

Figure 3.17: boxplots (see appendix 1) before and after global normalization.

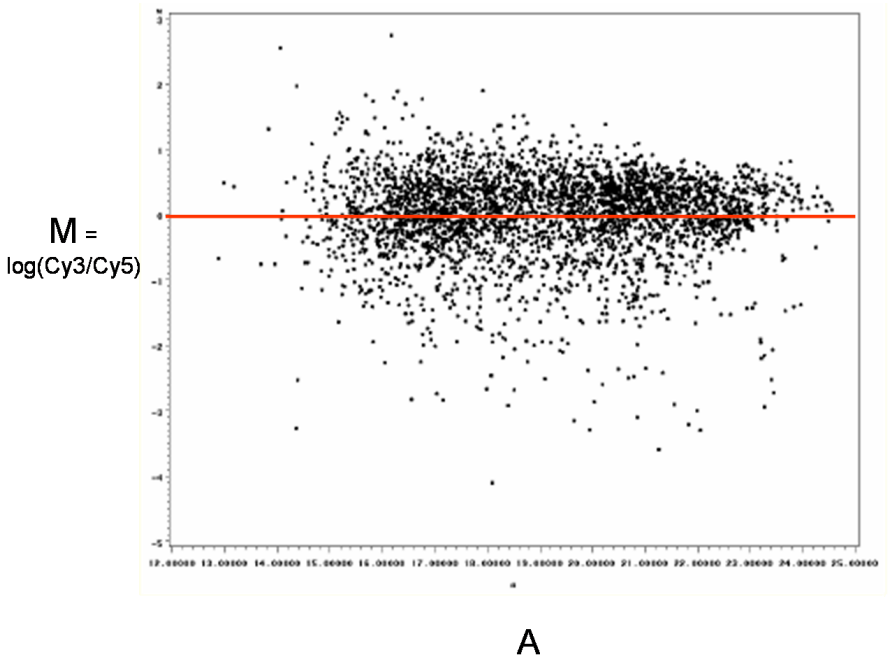

Figure 3.18: log intensity (M) vs. Average intensity (A) with no bias.

levels themselves (X axis)) On the other hand, Figure 3.19 shows a situation in which the ratio between expression levels changes completely for different expression levels. For lower expression levels one of the chips' values are measured to be higher than the other's, and this situation is reversed for higher expression values. It is obvious that this situation cannot be corrected by global normalization. Lowess normalization fits a local regression curve to the

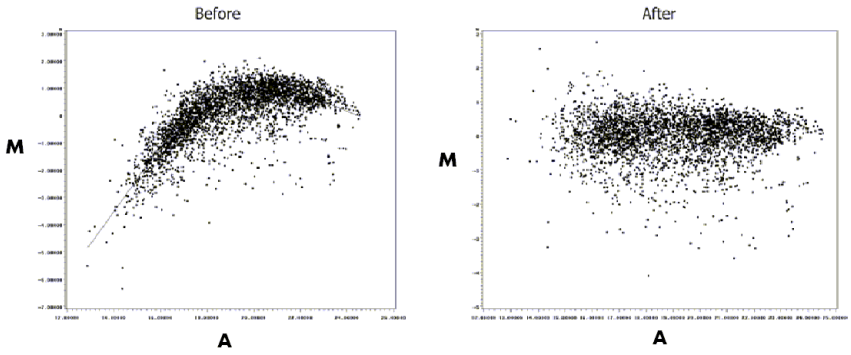

Figure 3.19: M vs. A with bias. Notice that on the left figure a global factor will not work, thus the Lowess method (right figure) is needed.

M vs. A graph and uses it to calculate a normalization factor that depends on the mean intensity. The normalization is performed by multiplying each gene expression level by the factor fitting its expression level (see Figure 3.19).

3. Quantile normalization

Contrary to the other normalization methods which tried to equalize the mean expression level. The first, is global normalization where a constant factor is used for the entire data as in Figure 3.16. The second, is the Lowess method where for each intensity level a local factor is used, as seen in Figure 3.20). *Quantile normalization* forces the chips to have identical intensity distributions (see Figure 3.21). The idea is to make sure that both chips will have the same intensity distribution histogram Of course, it doesn't promise that the same **genes** will have the same intensities, only the same distribution of intensities. Quantile normalization is done by sorting the gene expression levels. let E_i^j be the expression level of gene i in chip j. After sorting, let $\hat{E}_k^j$ be the k-th largest expression level for chip j. Of course, this is the expression level of gene i for some i : $\hat{E}_k^j = E_i^j$. We now compute the median intensity for each rank:

$$< I_k >= \frac{\sum \hat{E}_k^j}{T}$$

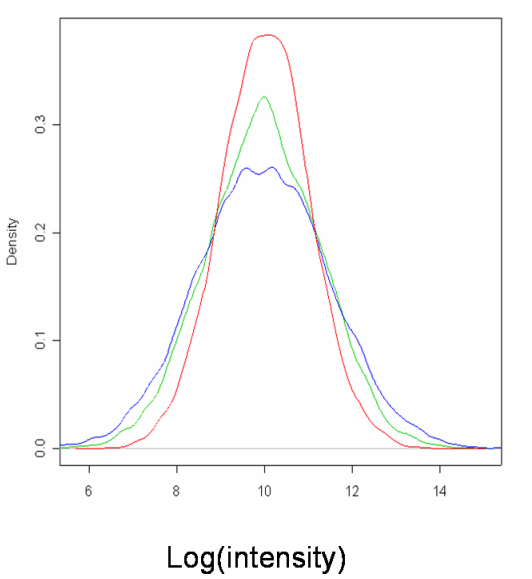

Figure 3.20: After Lowess normalization. Notice that the mean of all intensities distributions are the same, although the distributions themselves can be very different.

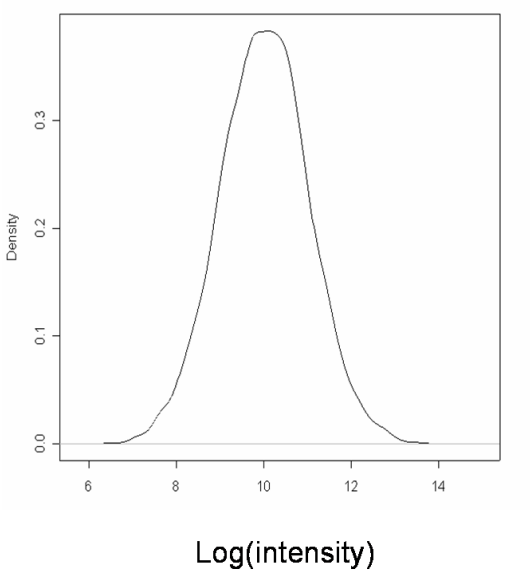

Figure 3.21: After quantile normalization. All distributions are identical.

Now we should normalize by replacing each gene i with expression level $\hat{E}_k^j$ with this median. In this way we make sure that for each rank k, we will have an expression level on each chip with the same value, thus the chips will have the same expression level distribution. (see Figure 3.22)

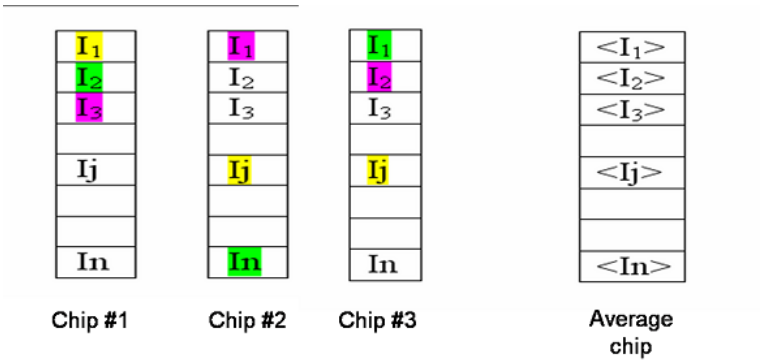

Figure 3.22: Quantile normalization. Each color is a specific gene, I_i denotes the ranked intensity of a gene. The I_i values are the average for each rank.

Summary

It appears that Quantile normalization is the best normalization method. Lowess has comparable results, but Global normalization is not satisfactory (The comparisons are based on the specific data sets used in experiments [4], [13]. Different data sets might give different results). There are a number of normalization tools available:

- BioConductor can be used on both Affymetrix and cDNA microarrays
- dCHIP can be used only for Affymetrix and is based on Quantile normalization, using the Invariant set method to choose normalization genes
- Expander can be used on both Affymetrix and cDNA microarrays and can use both Quantile normalization and Lowess.

3.2 Identification of Differential Genes

The most common microarray experiment is a comparison between 2 samples - a treatment sample and a control sample. The goal to identify genes that are differently expressed in the two samples. The number of microarrays is usually very low (2-4). There are a number of methods to identify the differently expressed genes. An important perquisite of these methods is the ability to asses the chance of *false positives*[6]. Without it, it is impossible to know whether the results of the experiment are reliable.

3.2.1 Fold change

In this method, all of the genes with expression level change (between treatment and control samples) of more then a given percentage (e.g. 75 - 100 percent) are treated as differential genes. This naive method has a number of major limitations:

- No estimation is given for the chance of false positives
- This method is biased toward lower expression genes, because for those genes, even a small change due to an error could be enough to mark them as differential. (see Figure 3.23) This situation could be improved by using a cutoff to filter genes with a too low expression level.
- There is no consideration of the variability of gene expression levels over a number of microarrays. For example, it is enough for one treatment microarray to show a very high expression level for a gene, for this gene to be marked as differential. Yet, in other treatment microarrays this genes might have low expression level, possibly of some other biological phenomena in the specific sample analyzed by the first microarray. (see Figure 3.24 for example)

Note that empirical results show a *false positive* of 60-70 percent using this method. We should seek a score that "punishes" genes with high variability over replicates.

3.2.2 t-test

The t-test is based on normalizing the expression level change, with the variance of the mean expression levels (of the treatment and control samples). In case the expression level change is large in comparison with the variance of the mean expression values, one can assume there is a real difference in gene concentration, i.e. the gene is differential. On the other hand, even

[6]The chance that a gene will be detected as differential even though it isn't one

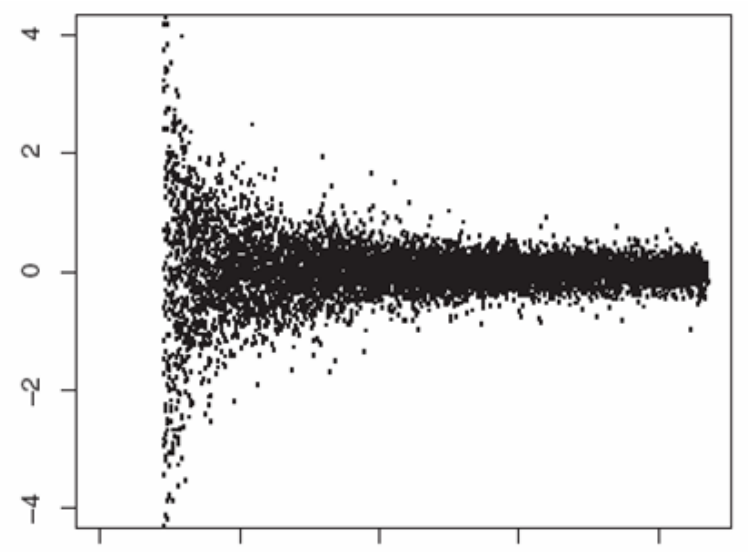

Figure 3.23: Fold Change limit. Here we see the fold change (Y-axis) as a function of the intensity (X-axis). Biased to low expression levels. Notice that for small values, a large fold change occurs even for a small change. In such situation one should consider choosing some cut-off intensity level, to avoid "noise" from the low intensity areas.

	control					treatment			
	C1	C2	C3	mean_c		t1	t2	t3	mean_t
g1	90	100	110	100		190	200	210	200
g2	50	100	150	100		100	150	350	200

Figure 3.24: Fold Change limit. Note that both g1 and g2 have the same mean value of 2 (200/100), however their variances are different. Eventually, they end up with the same t-score.

if the difference is large, but the gene has high variance, we will not treat it as differential. The *t-score* value is computed the following way:

$$t = \frac{M_c - M_t}{\sqrt{\frac{S_c^2}{n_c} + \frac{S_t^2}{n_t}}}$$

while S_c^2, S_t^2 are the variance estimates in control and treatment samples respectively. M_c, M_t are the mean levels in control and treatment samples respectively. n_c, n_t are the number of control and treatment samples respectively. It is possible to calculate a p-value[7] for each t-score in order to asses the chance for a false positive (the chance is the p-value itself). We should dismiss genes with p-values higher than some cutoff bound (see Figure 3.25). There

	C1	C2	C3	mean_c		t1	t2	t3	mean_t	t	p-val
g1	90	100	110	100		190	200	210	200	12.2	0.0001
g2	50	100	150	100		100	150	350	200	1.3	0.14

Figure 3.25: Example of computation of t-score and p-value when comparing control and treatment. Though, the fold change of both genes is 2, we can see that they have very different p-values. If we would consider only genes with p-value less than 0.01, only g1 would be declared differential

are other methods of estimating the p-value of difference between samples. One such method (that improves the variance estimation in case of a small number of tests) is *Cyber-T* ([1]).

3.2.3 Multiple Testing

t-score based methods are problematic when used for microarray analysis. The reason is the known statistical problem of *multiple testing*. When testing for a very large number of cases (genes), one should take into account the fact that the number of false positives might be high. For example, when looking at a specific case (gene), with p-value 0.0001 the probability for it not being significant (differential) when checking 10000 cases is quite high. Thus, if one wants to avoid receiving too many false positives the decision about the cut-off p-value should take into consideration the number of cases examined. This poses a question on the validity of the microarray results.

[7] the chance to have a given t-score in case of a random sample

Bonferroni correction

The Bonferroni correction([3]) states that in order to have a given chance of false positives q, while doing N experiments, one should aim for a p-value that is $\frac{q}{N}$. This follows immediately if one assumes that each test result is independent. For example, given the numbers described above, one should choose cutoff of 0.000001 p-value in order to have a chance of 0.01 for **one** false positive.

The problem with the Bonferroni correction is that the t-value required for such a low p-value will most probably limit the number of true positives found. In summary, using the Bonferroni correction promises a low chance for false positives but also may cause a large number of false negatives (differential genes that would be filtered because of the high t-value threshold).

False Discovery Rate

The idea behind *false discovery rate (FDR)*([2],[9]) is to choose an acceptable proportion of false positives among the genes declared as differential, for example 10 percent (this percentage will be marked q). The FDR method is to rank the tested genes according to their p-values and choose as differential genes, only the first k genes, those with the lowest p-value so that

$$p_i \leq i * \frac{q}{N}$$

so we will guarantee that the false positives amount is not exceeded.

The problem with FDR is that it, like the rest of the presented methods, assumes that the gene expression of different genes on the chip is independent. This is biologically incorrect - many genes' expressions are correlated.

Significance Analysis of Microarray

Significance Analysis of Microarray (SAM)([11]) is intended to deal with the fact that gene expressions are correlated in an unknown manner. The idea is to use permutations to get an 'empirical' estimate for the FDR of the reported differential genes. Instead of using the above FDR calculation, one tries to *rename* the different genes as if the two sample groups have been mixed up (e.g. we take 3 "green" control samples and 3 "red" treatment samples and change their colors). We take many different permutations and by summing up the resulting number of differential genes we can understand the significance of the original result. The lower the number of differential genes we get under a random permutation the higher the chance that our result is true. The SAM algorithm is :

- Compute for each gene a statistic that measures its relative expression difference in control vs 'treatment' (t-score or a variant)
- Rank the genes according to their 'difference score'
- Set a cut off d_0 and consider all genes above it as differential. the number of differential genes is N_d.
- Permute the condition labels, and count how many genes got score above d_0. The number of genes is N_p
- Repeat on many (all possible) permutations and count N_{pj}
- Estimate FDR as the proportion: $\frac{<N_{pj}>}{N_d}$

3.3 Appendix

3.3.1 Boxplots

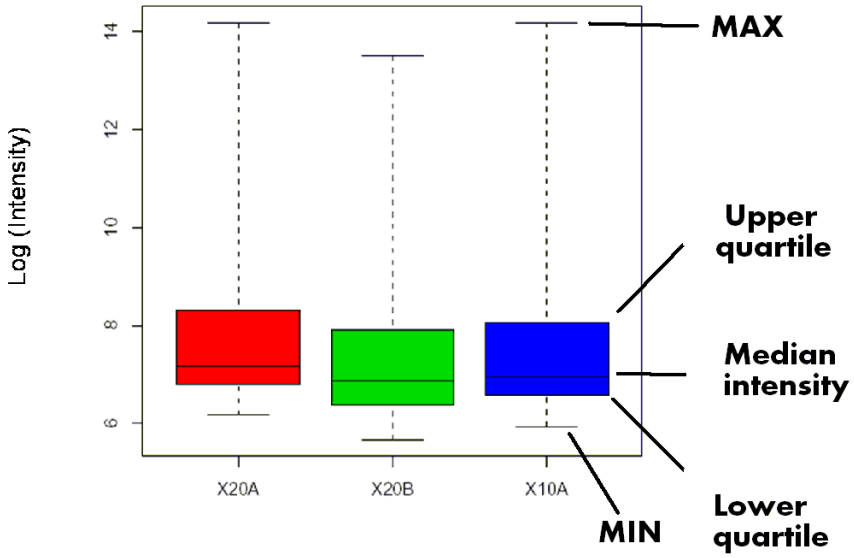

Figure 3.26: Explanation of boxplots diagrams

Boxplots are method for graphical representation of a distribution, based on representing the different quartiles. The range is divided by five values (as shown in Figure 3.26):

- The upper line indicates the maximal value.
- The upper line in the colored box indicates the upper quartile of the values.
- The middle line in the colored box indicates the median.
- The lower line in the colored box indicates the lower quartile of the values.
- The lower line indicates the minimal value.

The five number summary leads to a graphical representation of a distribution called the boxplot.

Bibliography

[1] P. Baldi and A. D. Long. A bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes. *Bioinformatics, 17: 509-519*, 2001.

[2] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: A practical and powerfull approach to multiple testing. *J.R Statist. soc, Ser B. 57: 289-300*, 1995.

[3] C. E. Bonferroni. Il calcolo delle assicurazioni su gruppi di teste. *Studi Onore del Professore Salvatore Ortu Carboni, 13-60*, 1935.

[4] B.M. Bolstad *et al. A comparison of normalization method for high density oligonucleotide array data based on variane and bias.* Bioinformatics, 19(2):185-93*, 2003.*

[5] R. A. Irizarry et al. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics, 4(2):249-64*, 2003.

[6] R. A. Irizarry *et al. Summaries of affymetrix genechip probe level data.* Nucleic Acids Res, 31(4):e15*, 2003.*

[7] Y. H. Yang et al. Normalization for cdna microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Res, 30(4):e15*, 2002.

[8] C. Li and W. H. Wong. Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *PNAS, 98:31-36*, 2001.

[9] V. Melfi. False discovery rates and their application to microarray data analysis. *http://www.stt.msu.edu/ huebner/melfifdr.pdf*, 2003.

[10] G. K. Smyth and T. Speed. Normalization of cdna microarray data. *METHODS: Selecting Candidate Genes from DNA Array Screens: Application to Neuroscience, 29-37*, 2003.

[11] V. Tusher., R. Tibshirani., and G. Chu. Significance analysis of microarrays applied to the ionizing radiation response. *PNAS, 98: 5116-5121*, 2001.

[12] `http://research.nhgri.nih.gov/microarray/image_analysis.html`.

[13] `http://stat-www.berkeley.edu/users/bolstad/normalize/`.

[14] `http://www.dchip.org/`.